

Application No.: 10/720,096Docket No.: 200316136-2 (1509-476)**REMARKS**

Claims 21 and 39 have been canceled to render the objection to them moot. These claims have been amended because it has been recognized that the gcc 2.96 compiler has certain problems and should not be used. Claim 3 has been amended to overcome the rejection thereof under 35 USC 112, second paragraph. Some of the remaining claims have also been amended to define applicants' contribution to the art with greater particularity, in some cases, and in slightly broader breadth in other cases, such as in claims 4, 11, 28 and 31. The apparatus claims have been amended to insure infringement prior to the system being used in the United States. The subject matter of claim 12 has been incorporated into each of independent claims 1 and 25 and claim 12 has been canceled. Claim 50 has been added to provide a basis for infringement of the combination of the non-data relevant environment and the system of claim 25.

Applicants traverse the rejection of claims 1-49 under 35 USC 112, second paragraph, which alleges independent claims 1 and 25: (1) do not properly explain "who is locating one or more instances"; (2) "what information locating instances of one or more data-relative offset in code segment occurs"; (3) "who is calculating a new offset... and based on what information"; and (4) "who is replacing data-relative offset with the new offset (sic)." In response, claim 25 requires a processor arrangement adapted to perform locating, calculating and replacing operations. Consequently, claim 25 indicates the structure for performing the above operations (1), (3) and (4). Applicants do not understand how the above operation (2) is applicable to claim 25.

Claim 1 is a method claim. As such, there is no need to define the specific structure for performing the locating, calculating and replacing steps. There is no requirement under 35 USC 112, second paragraph, to indicate what information is

Application No.: 10/720,096Docket No.: 200316136-2 (1509-476)

required to meet the limitation of claim 1 to locate "one or more instances of the use of a data-relative offset within a module of a code segment." The primary purpose of 35 USC 112, second paragraph, is to ensure that the scope of the claims is clear so the public is informed of the boundary of what constitutes infringement of the patent. A secondary purpose is to provide a clear measure of what applicants regard as the invention so that it can be determined whether the claimed invention meets all criteria for patentability and whether the specification meets the criteria of 35 USC 112, first paragraph with respect to the claimed invention. MPEP Section 2173. The examiner has failed to show why the foregoing purposes of 35 USC 112, second paragraph, are not satisfied by the language of claim 1. 35 USC 112, second paragraph, does not require a step to be explained. Breadth of a claim is not to be equated with indefiniteness. In re Miller, 441 F.2d 689, 169 USPQ 597 (CCPA 1971). If the scope of the subject matter embraced by the claims is clear and if applicants have not otherwise indicated they intend the invention to be of a scope different from what the claims define, the claims comply with 35 USC 112, second paragraph. Based on the above, the rejection of all claims based on 35 USC 112, second paragraph is erroneous.

Amended independent claims 1 and 25 are not anticipated by Peterson, US patent 5,504,901, despite the allegation in the office action that Peterson discloses the subject matter of claim 12 that is now included in claims 1 and 25. Claims 1 and 25 now require the locating, calculating and replacing operations or steps to occur outside the runtime of an application using the code segment. The office action alleges, in paragraph 15, page 5, this requirement is disclosed by Peterson at column 6, line 60-63. In fact, this portion of Peterson refers to loading procedures and data sets into memory portions identified by an initial or starting address indicated by arrows 50, Figure 1. Nothing is stated in this portion of Peterson about locating, calculating and replacing operations or steps occurring outside the runtime of an application using the code segment. To the contrary, Peterson indicates the opposite occurs. For example, column 7, lines 57-62 states: "When the procedure calls a different procedure for invocation and execution by the central processing unit, execution of the calling

Application No.: 10/720,096**Docket No.: 200316136-2 (1509-476)**

procedure call sequence instructions causes the entry point address of the call procedure to be transferred (passed) to the predetermined memory location (e.g., general register). Similarly, column 9, lines 30-45 indicates that, if during the execution of procedure 20, a call sequence is encountered that invokes yet another procedure, the value stored in procedure entry point register 66 and additional offset values are utilized to invoke and begin executing the procedure being called by procedure 20. The example is given that if call sequence 60 of procedure 20 in Figure 1 calls procedure 24 of image file 28, the instructions of call sequence 60 utilize the value stored in procedure entry point register 66 (that is, the entry point address 50 of procedure 20) and offset pointer values to locate a memory field 80 of linkage code 72 holding the address of procedure 24. Based on the foregoing, in Peterson, the calculating and replacing steps occur inside the runtime of an application using the code segment, rather than outside the runtime of an application using the code segment, as now required by claims 1 and 25.

Because claims 1 and 25 distinguish over Peterson, and the Lee et al. reference, US patent 5, 828, 884 (relied on to disclose the features of claims 8 and 42, and used in combination with Peterson to reject claims 8 and 42) does not disclose or make obvious the foregoing limitation of claims 1 and 25, the remaining claims are patentable over the art of record. In addition, many of the remaining claims are improperly rejected under 35 USC 102(b) or 35 USC 103(a).

Claims 7 and 41 require an HP-UX platform, and claims 9 and 43 require a non-native platform. The office action alleges the features of these claims are disclosed by Peterson at column 13, lines 6-10. However, column 13, lines 6-10 makes no mention of either an HP-UX or a non-native platform. This portion of Peterson merely indicates the Peterson device can be used in a wide range of data processing environments and is sufficient both with respect to memory overload and execution time. A proper rejection under 35 USC 102 (b) requires a single reference to disclose all features of a claim. It is not sufficient for a reference to include a broad statement.

Application No.: 10/720,096**Docket No.: 200316136-2 (1509-476)**

The office action relies on column 12, lines 1-7 of Peterson to disclose the requirement of claims 10 and 30 for the new offset to be an absolute runtime address. There is no mention of a new offset being an absolute runtime address in column 12, lines 1-7 of Peterson and the examiner has offered no rationale or proof to show that the address calculation discussed in column 12, lines 1-7 inherently discloses a new offset as being an absolute runtime address. A proper rejection based on inherency requires such a showing by the examiner.

The office action relies on column 4, lines 27-30 of Peterson to disclose the requirements of claim 15 for the step of copying the code segment module to modifiable memory to occur before the replacing step of claim 1, upon which claim 15 is ultimately dependent. Column 4, lines 27-30 of Peterson merely indicates the system executes instructions that call another procedure, wherein the address of the called procedure is transferred (passed) to the procedure entry point register, identified in the detailed description of Peterson as being register 66. The examiner has provided no rationale or evidence to support the position that column 4, lines 27-30 of Peterson copies the code segment module to modifiable memory before the replacing step of claim 1.

To reject the requirement of claim 16, which depends on claim 15, the examiner relies on column 6, lines 8-13 of Peterson. Claim 16 includes the step of allocating the modifiable memory (of claim 15) read, write and execute permissions prior to the copying step of claim 15. Column 6, lines 8-13 indicates the processing task defined by procedures 18, 20, 24 and 26 are performed by central processing unit 12 communicating with memory 14 via addressing and data bus 16 to sequentially access or fetch coded instructions of the procedure being executed and perform the function specified by the instructions. It is not seen how this disclosure satisfies the requirements of claim 16; explanation is requested.

Claim 17, dependent on claim 14, requires the code segment within a shared

Application No.: 10/720,096**Docket No.: 200316136-2 (1509-476)**

library to be mapped as writable. The office action alleges Peterson discloses this feature at column 19, lines 28-31. Column 19, lines 28-31 states that if the object file or a shared library procedure being linked supports the practice of the invention, the linker modifies the position dependent code called sequences of the object file or library procedure. It is also not seen how this disclosure satisfies the requirements of claim 17.

The limitations of claim 18, requiring the code segment module to be selected from the set of an .init section from the shared library and a .fini section from the shared library, are stated to be disclosed at column 18, lines 27-30. Column 18, lines 27-30 indicates that, as the object code of each object file in each shared library is processed, a determination is made as to whether the object file or shared library must be safe for use with the position independent code of the Peterson reference. The examiner is requested to indicate how this portion of the reference discloses the foregoing requirements of claim 18.

The office action alleges column 4, lines 7-9 of Peterson discloses the requirement of claims 24 and 40 for the code segment to be a Linux code segment. The relied on portion of the reference merely indicates programs written for use with position dependent code can be used as program parts, and a process that utilizes position independent code can be employed with the Peterson arrangement. Because there is no mention of a Linux code segment in column 4, line 7-9 of Peterson, this rejection based on 35 USC 102 (b) is incorrect.

The office action alleges the requirements of claims 4, 11, 28 and 31 are unpatentable under 35 USC 103(a) because of Peterson. The office action says the equation Peterson discusses at column 10, line 42-column 11, line 15 can be mathematically manipulated to meet the requirements of claims 4 and 28. The office action also alleges the equation Peterson discusses at column 10, line 42-column 11, line 15 can be mathematically manipulated to meet the requirements of claims 11 and 31 that differ from the requirements of claims 4 and 28. The examiner is requested to

Application No.: 10/720,096

Docket No.: 200316136-2 (1509-476)

indicate the alluded to mathematical manipulations for the expressions of both claim sets.

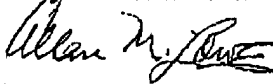
In view of the foregoing amendments and remarks, allowance is in order.

To the extent necessary, a petition for an extension of time under 37 C.F.R. 1.138 is hereby made. Please charge any shortage in fees due in connection with the filing of this paper, including extension of time fees, to Deposit Account 08-2025 and please credit any excess fees to such deposit account.

Respectfully submitted,

LOWE HAUPTMAN HAM & BERNER, LLP

Manoj N. PALAT et al.



Allan M. Lowe
Registration No. 19,641

HEWLETT-PACKARD COMPANY
Intellectual Property Administration
P. O. Box 272400
Fort Collins, CO 80527-2400
703-684-1111 Telephone
970-898-0640 Telecopier
Date: January 18, 2008
AML/cjf